Microprocessors and Microsystems 23 (1999) 123–124

Editorial

Engineering complex computer systems

A. Broggi*, M. Hinchey, A.D. Stoyen

Dipartimento di Informatica e Sistemistica, Università di Pavia, Via Ferrata 1, I-27100 Pavia, Italy

Received 1 January 1999; received in revised form 13 April 1999; accepted 21 May 1999

Surely, nobody could have foreseen the degree to which we have become reliant on computer systems in our daily lives. They wake us up in the morning, cook our food, get us to work, control the traffic in the streets, entertain us, give us cash from our bank accounts (under certain conditions!), help to treat our diseases, monitor our health, and even defend us from potential threats. The increases in productivity and greater precision that they afford has meant great advances in scientific discovery.

Yet, there has always been a naivety about how reliant we would be on computer systems. Back in the early 1950s, a survey by the British government estimated that two computers of the power and capacity of EDSAC (the first electronic computer to execute a stored program; built at the University of Cambridge and first run on 7 May 1949) would be sufficient to support all the scientific computing in the country. That was clearly a great underestimate, as the machine this introduction is being typed on, is many thousands of times more powerful than EDSAC and yet frustrates sometimes with long delays.

Similarly, we tend to forget that computer systems evolve over time and are often relied upon to work efficiently and correctly many years after their "sell-by date". An example of this is the so-called Millennium bug (or Y2K problem). It is not a bug; it is not some sort of an inherent phenomenon—it is merely that some 30 years ago the US Navy took a decision to represent years in COBOL programs by two digits, not foreseeing the length of time that computer systems would be running. Legacy systems have become ubiquitous—often we must rely on computer systems or components of computer systems that were written many years previously, where none of the original developers are available to help with system maintenance and where little or no documentation is available.

But perhaps the greatest naivety is the belief that large-scale, complex computer systems that must operate within strict parameters, such as timing constraints, satisfy government and other regulations, be reliable and dependable can be built in the same manner as more simple systems. The difference is far from being merely one of scale. Just because we can add to our home PC system by purchasing new software and hardware components from a computer retail store, it does not follow that we can modify and extend large-scale systems in a similar manner, using COTS components, etc. Adding 10 times as many processors to a system does not make it 10 times more powerful, but it does add many opportunities for the system to fail to operate as we anticipated.

The development of large-scale complex systems cannot be approached in an ad hoc manner. There are a whole plethora of issues to consider—timely performance, reliability, dependability, security, safety, efficiency, cost, and maintainability to name just a few. Unfortunately, optimizing one aspect can often have a negative effect on another aspect. In fact, the engineering of complex computer systems has become a research issue in its own right. For three years now we, under the auspices of the IEEE Computer Society Technical Committee on Complexity in Computing, have organized a special mini-track at the IEEE Hawaii International Conference on System Sciences (HICSS) on the topic of Engineering Complex Computer Systems.

We have found that there is great interest in this area and have been successful in attracting many excellent contributions for the mini-track, several of which have been candidates for the best paper award. This year was no exception, and the five papers in this special issue are revised versions of papers presented during the mini-track at HICSS held last January.

The topics covered are varied, but all have an emphasis on reducing and conquering complexity in large-scale computer systems. Papers in this special issue discuss real-time systems, parallel processing, parallel algorithms, high performance computing, analysis, and more.

Houzet described how clusters of PCs might use a shared memory model. Gebotys and Gebotys described how embedded digital signal processing systems could be designed for effective use in low-power situations. Chen et al. discussed real-time support in COM, real-time being

* Corresponding author. Tel.: +39-0382-505756; fax: +39-382-50-5373.

E-mail address: broggi@dis.unipv.it, web: http://www.ce.unipr.it/people/broggi (A. Broggi)

0141-9331/99/$ - see front matter
PII: S0141-9331(99)00034-4

one of the areas that offers many great challenges. Park and George describe parallel algorithms for string matching based on dataflow, an area that is increasingly attracting attention. Finally, Hariri et al. described a hierarchy-based approach for the analysis of applications in high performance computing and communications.

We hope that these papers will hold something of interest for you. We would like to thank each of the authors for their contributions both at HICSS and for their revised contributions here. Our sincere thanks also to the publishers and editors of this journal for the opportunity to bring the Engineering of Complex Computer Systems to a wider audience.